分层边缘计算中利用联邦学习的差分隐私保护

董少华[1,2],马新春[2],樊小超[1]
(1. 新疆师范大学 计算机科学技术学院,新疆 乌鲁木齐 830054;
2. 新疆电子研究所,新疆 乌鲁木齐 830013)

摘 要:分层边缘计算通过在资源受限的边缘设备和云服务器上协作运行,旨在最小化推理延迟和保护数据隐私。然而,即使来自边缘设备的原始输入数据没有直接暴露于云中,针对边缘数据的最先进的攻击仍然能够从暴露的本地模型的中间输出中重建原始的私有数据,从而引入严重的隐私风险。为了克服这些限制,提出了一种新的算法,该算法合并了边缘阶段和云阶段的模型,提出了确定最佳聚合时间框架的定性指令,以减少计算和通信费用。通过在客户端和边缘服务器级别实现局部差分隐私,增强了本地模型参数更新时的隐私。在 CIFAR-10 和 MNIST 数据集上的实验表明,该算法在训练精度、训练时间和通信-计算权衡方面优于标准的联邦学习方法。且该算法为分层边缘计算的挑战提供了一个很有前途的解决方案,使内容交付更快、移动服务质量更高。

关键词:分层边缘计算;联邦学习;差分隐私;边缘协作;数据隐私
中图分类号:TP391.1 **文献标识码**:A **文章编号**:1673-629X(2025)04-0053-06
doi:10.20165/j.cnki.ISSN1673-629X.2024.0382

Differential Privacy Protection Using Federated Learning in Layered Edge Computing

DONG Shao-hua[1,2], MA Xin-chun[2], FAN Xiao-chao[1]
(1. School of Computer Science and Technology, Xinjiang Normal University, Urumqi 830054, China;
2. Xinjiang Electronic Research Institute, Urumqi 830013, China)

Abstract: Layered edge computing, by operating collaboratively on resource-constrained edge devices and cloud servers, aims to minimize inference latency and protect data privacy. However, even if the raw input data from edge devices is not directly exposed to the cloud, sophisticated attacks targeted at edge data can still reconstruct the original private data from exposed local model outputs, introducing serious privacy risks. To overcome these limitations, a new algorithm is proposed that amalgamates models from both the edge and cloud phases, presenting qualitative directives for determining optimal aggregation time frames to reduce computational and communication costs. Local differential privacy is implemented at the client and edge server levels to enhance privacy during local model parameter updates. Experiments on the CIFAR-10 and MNIST datasets demonstrate that the proposed algorithm outperforms standard FL methods in terms of training accuracy, training time, and communication-computation trade-offs. It offers a promising solution to the challenges of layered edge computing, enabling faster content delivery and higher mobile services quality.

Key words: layered edge computing; federated learning; differential privacy; edge collaboration; data privacy

0 引 言

随着物联网技术的迅速发展,物联网设备也得到了迅速普及,使得不同数据的生成大幅上升,不仅包括自动化终端,还包括生产设备和检测设备等智能终端。获取数据隐私保护相关知识的复杂性,以及将所有数据传输到远程云的高昂成本,促进了一种潜在的解决方案——分层边缘计算(Layered Edge Computing, LEC)的出现。分层边缘计算技术的采用提供了一些潜在的好处,包括提高能源效率和降低延迟[1]。尽管如此,这种方法也提出了一些与隐私相关的担忧,如无意中披露敏感的用户信息、拦截经过训练的模型,以及文献[2]算法中监管框架不足等问题。因此,一种稳健和全面的隐私保护和治理方法对于分层边缘计算技术的可持续扩散至关重要。

收稿日期:2024-08-23 **修回日期**:2024-12-25
基金项目:新疆维吾尔自治区重点研发计划项目(2022B01008);天山英才培养计划项目(2023TSYCJCO039)
作者简介:董少华(1999-),男,硕士研究生,通讯作者,研究方向为边缘计算和自然语言处理;马新春(1967-),男(回族),正高级工程师,硕导,研究方向为物联网、边缘计算等;樊小超(1982-),男(锡伯族),副教授,硕导,研究方向为自然语言处理、情感分析等。

为了解决上述类似问题以及文献[3]中优化沟通后造成潜在的数据泄露等隐私问题,联邦学习(Federated Learning,FL)通常被认为是一个有效的框架。这是因为它能够促进使用本地数据的学习程序,同时也能保存敏感信息。传统的 FL 框架允许用户在本地数据上训练他们的学习模型,然后他们只将必要的训练权值发送到中央服务器进行聚合。一旦本地模型被聚合,更新后的全局模型将广播给用户进行进一步细化。虽然 FL 为用户的隐私提供了保护,但它需要大量的通信和模型培训资源,这对于低成本的设备来说可能是相当具有挑战性的。将模型更新迭代传输到中央云服务器的过程创建了大量的通信负载,这对管理相当具有挑战性。此外,复杂深度神经网络(DNN)的训练对内存和计算资源[4]有很大的需求。这些障碍可能会阻碍 FL 的广泛合并,特别是对于资源有限的边缘客户端。

为了解决 FL 中客户的资源限制问题,提出了分裂 FL 的概念。分裂联邦学习的关键好处是,它允许在移动设备上学习详细设计的模型,而不会加重其负担。分裂联邦学习方法涉及到将机器学习(Machine Learning,ML)模型划分为几个部分,其中一个部分被指定为支持远程服务器训练。客户机参与其模型的传播,同时将中间结果传输到远程服务器,以促进模型培训的完成。与 FL 相比,在分裂联邦学习的训练策略中加入并行处理和网络分裂,减少了客户端的内存和处理需求,尽管仍然需要解决一定的限制。与 FL 面临的挑战不同,部分结果频繁传输到远程服务器造成了重大障碍,因为它放大了通信资源的使用和传输时间,并要求采取措施保证传输过程中中间结果的机密性。

自引入文献[5]的分裂学习以来,分裂联邦学习(Splitting Federated Learning,SFL)经历了不断的发展,并已被应用于各种研究工作中。文献[6]研究检查了两种学习架构,它们将 FL 和分裂学习(Split Learning,SL)合并,以减少客户端计算需求和并行化 SL。相比之下,文献[7]在无人机网络中实现了分裂联邦学习,以解决数据传输和隐私问题。文献[8]还强调了在 u 型医学图像网络中的分裂联邦学习应用。然而,分裂联邦学习面临着一些挑战,包括中央云服务器紧张、通信延迟和隐私问题。分裂联邦学习依赖于一个中央服务器聚合来自所有客户端的更新[9]。这可能会导致瓶颈和性能下降,特别是在许多客户端中。此外,在每次迭代中,所有客户端都必须与中央服务器进行通信,这可能会导致通信延迟,特别是对于远程客户端或拥塞的网络。最后,分裂联邦学习要求客户端在没有隐私保护技术的情况下经常向其本地模型更新发送到中央服务器,导致出现隐私问题,因为在服务器和客户端之间的通信期间可能会发生攻击[10]。

为了解决这些挑战,该文提出了算法 ECDPFL,这是一个新的框架,利用分层边缘服务器作为培训助手和模型聚合器。在 ECDPFL 中,客户端被分成几组并分配给分层边缘服务器(Layered Edge Servers,LESs)。每个组中的客户机不需要将数据发送到中央服务器;相反,它们在将数据发送到指定的分层边缘服务器之前,会使用本地差分隐私(Local Differential Privacy,LDP)向数据添加噪声。然后,分层边缘服务器从其分配的客户端聚合更新,并将聚合后的更新发送到中央服务器。中央服务器更新全局模型并将其发送回分层边缘服务器,分层边缘服务器将其分发给指定的客户端。这种方法显著减少了中央服务器的负载并改善了通信延迟,因为客户端只需要与分配的分层边缘服务器通信,后者通常靠近它们。该文的贡献可以总结如下:

(1)提出了一个在分层边缘服务器和云级别上都具有模型聚合的分层分裂联邦学习框架,其中开发了定性指南来确定每个级别上的最佳聚合间隔。这有助于平衡计算和通信成本。

(2)在两个客户端级模型上都实现了 LDP,以增强在本地模型参数同步期间的机密性。

(3)在 CIFAR-10 和 MNIST 数据集上进行了实验,证明了 ECDPFL 方案优于传统的 FL 方法,具有更好的通信计算权衡。

1 相关工作

1.1 差分隐私

差分隐私(Differential Privacy,DP)是由 Dwork 在 2006 年提出的,作为一种可靠的隐私模型,近年来被认为是机器学习中一种很有前途的隐私保护策略。DP 的定义是基于一个严格的理论基础,隐私保护在机器学习是通过添加可量化的噪声模型或输出结果实现可证明的机制。并通过在一个优化的权衡隐私强度和可用性之间进行调整来实现隐私预算。

差异隐私保证了相邻两个数据集上任意随机算法的查询和访问具有相似的输出分布,攻击者不能在任何查询结果中推断出个体的私有信息。差别隐私的正式定义如下:

(1)对于任意两个相邻的数据集 D 和 D',给定一个运行在这个相邻数据集上的随机机制 M 算法,当以下不等式成立时,S 是由 M 机制产生的所有可能输出的子集。

$$p_r[M(D) \in S] \leqslant e^{\varepsilon} p_r[M(D') \in S] \tag{1}$$

然后通过被称为随机机制 M 来满足 ε -差分隐

私,其中参数 ε 为隐私保护预算, ε 值越小,隐私保护强度越高。

(2)在介绍具体的隐私保护机制之前,需要给出全局敏感性的定义。它是使用查询函数 $f:\mathbb{N}^{|x|}\rightarrow\mathbb{R}^k$ 对任意两个相邻的数据集 D 和 D' 进行全局灵敏度定义。

$$\Delta s(f,\|\bullet\|)=\max_{d(D,D')=1}\|f(D)-f(D')\| \tag{2}$$

其中, $\|\cdot\|$ 是距离度量,通常采用 l_1 和 l_2 范数。

基于上述差分隐私和全局灵敏度的定义,拉普拉斯机制通过在查询结果中添加服从拉普拉斯分布来实现 ε -差分隐私保护机制。

(3)给定任何查询函数 $f:\mathbb{N}^{|x|}\rightarrow\mathbb{R}^k$,具有全局 l_1 灵敏度 Δs ,拉普拉斯机制定义为:

$$M_l((x,f(\bullet),\in))=f(x)+(Y_1,Y_2,\cdots,Y_k) \tag{3}$$

其中, Y_i 来自独立同分布 $\mathrm{Lap}(\frac{\Delta s}{\varepsilon})$ 的随机变量,且拉普拉斯机制满足 ϵ -差分隐私保护机制。

1.2　联邦学习

在 FL 中引入差分隐私(DP)主要遵循两种范例:(1)中央 DP(CDP),包括在主服务器上增加噪声;(2)本地 DP(LDP),它在每个边缘设备上增加噪声。CDP 通常会产生一个更准确的最终模型,但它取决于主服务器的可信度。相反,LDP 放弃了这种信任要求,但需要在每个设备上添加更高水平的噪声来补偿。

通常,实现 FL 的网络系统是分层的,即,使用一个或多个层的雾计算元素将边缘设备与云服务器分开。在此基础上,引入了第三种称为层次 DP(HDP)的范式。HDP 假设某些“超级节点”存在于计算节点的层次结构中(例如,边缘服务器、中间路由器等),即使主服务器不能信任,但它们依然是可信的。这些超级节点的任务是在传输前向聚合模型中添加经过校准的 DP 噪声。HDP 不是均匀地注入噪声,而是能够调整系统内不同信任级别的噪声。

已经有一些研究工作致力于将 DP 技术集成到分层的 FL 中。文献[11-12]都将 LDP 策略适应于分层的 FL 结构,利用时刻会计法在整个系统中获得严格的隐私保证。文献[13]探讨了灵活的分散控制在分层 FL 训练过程中的优势,并研究了其对参与者隐私的影响。文献[14]介绍了 HDP 的概念,强调了与 LDP 相比,由于在“超节点”级别注入 DP 噪声特性的隐私放大的机会。虽然取得了一些成效,但仍然存在缺陷。而在 LDP 的情况下,正式的收敛分析将为参数选择和必要的噪声注入提供更精确的指导,以实现理想的隐私-性能平衡。

1.3　分裂联邦学习

在 SFL 框架中, S_t 由 t 时刻的一组 k 个客户端组成。每个客户端对其带有噪声层的模型执行并行前向传播。然后,它们将压缩数据 $D_{k,t}$ 和标签 Y_k 发送到中央服务器。p_k 为客户端 k 的样本量, p 为总样本量。在培训期间,客户端 k 同时与 ρ_1 和 ρ_2 服务器进行交互。具体来说,在服务器 ρ_1 上,执行后续步骤:

(1)在带有压缩数据 $D_{k,t}$ 的全局服务器模型 w_t^s 上进行正向传播。

(2)通过正向传播计算预测的标签 $\hat{Y}_k$ 。

(3)使用 Y_k 和 $\hat{Y}_k$ 进行损失函数计算,计算公式如下:

$$L(w_t^s;D_{s,t})=\sum_{k=1}^{k}\frac{p_k}{p}\ell(Y_k,\hat{Y}_k) \tag{4}$$

其中, n 为总样本量, n_k 为客户端 k 的样本量。

(4)在服务器端模型上的反向传播过程中,从每个客户机中分离出压缩数据的并行处理。每个客户端接收压缩数据 $\nabla\ell_k(w_t^s;D_{s,t})$ 的梯度,用于其反向传播过程。

(5)服务器的模型通过 FedAvg 进行更新,这涉及到在对每个客户端的压缩数据进行反向传播时计算出的梯度的加权平均值。

$$w_{t+1}^s=w_t^s-\eta_t\frac{1}{p}\sum_{k=1}^{k}\frac{p_k}{p}\nabla\ell_k(w_t^s;D_{s,t}) \tag{5}$$

在接收到其压缩数据 $\nabla\ell_k$ (wst; d , t)的梯度后,每个客户端利用它们在其本地模型上进行反向传播,并推导出其梯度 $\nabla\ell_k$ (wck, t)。在被发送到服务器 ρ_2 之前,通过 LDP 机制保护这些梯度,服务器 ρ_2 执行客户端本地更新的 FedAvg,并将结果传播给所有参与的客户端以获取隐私。

$$w_{t+1}^s=\frac{1}{p}\sum_{k=1}^{k}\frac{p_k}{p}w_{k,t}^c \tag{6}$$

2　ECDPFL 模型构建

在本节中,讨论了 FL 和 SFL 的主要学习问题,并提出了一个三层 FL 系统 ECDPFL。它作为一个在分层边缘计算网络中启用 SFL 的解决方案,引入了增强 ECDPFL 隐私的 LDP 机制。

2.1　ECDPFL 框架

ECDPFL 是一种新的 ML 方法,利用了 FL 和 SL 的优点。虽然云参数服务器上的模型聚合可以容纳大量的客户,但它会产生大量的通信费用。相比之下,在 LEC 参数服务器上参与模型聚合的少量客户端可以大大降低通信费用。因此,提出了一个 ECDPFL 框架来获得这两种方法的优势。提出的框架包括一个云服

务器和 M 个 LESs,每个服务器都由索引 m 标识。LESs 服务于单独的客户端集,它们被标记为 $\{C^m\}_{m=1}^{M}$。此外,有 K 个客户端,由 k 和 m 共同索引,每个客户端都拥有分布式数据集 $\{D_k^m\}_{k=1}^{K}$。在每个 LES 下收集的数据集用 D^m 表示。每个 LES 都有责任促进协作培训和来自它所服务的客户的模型的聚合。

提出的 ECDPFL 方案的工作流程如图 1 所示。具体来说,从客户端到本地训练模型,中央云服务器使用客户端信息将客户端分配给 LESs。在从本地训练模型到分层边缘服务器协作培训步骤之前,客户先进行本地模型训练。然后,客户一起进行并行模型训练,保持 LES 更新中间结果,并接受边缘侧的辅助和梯度,以便进一步发展。这个循环发生在 E 次迭代中。其中,LESm 在对 C_m 集中的每个客户端进行每次 ρ_1 本地更新后,使用 FedAvg 算法聚合模型参数。边缘聚合过程在 ρ_2 轮询后结束,模型参数从 LESs 转发到云服务器。云服务器采用 FedAvg 算法将模型参数组合起来,并将其发送到 LESs,LESs 更新边缘模型并广播给客户端。ECDPFL 算法支持协同机器学习,同时保护数据隐私和最小化通信开销。

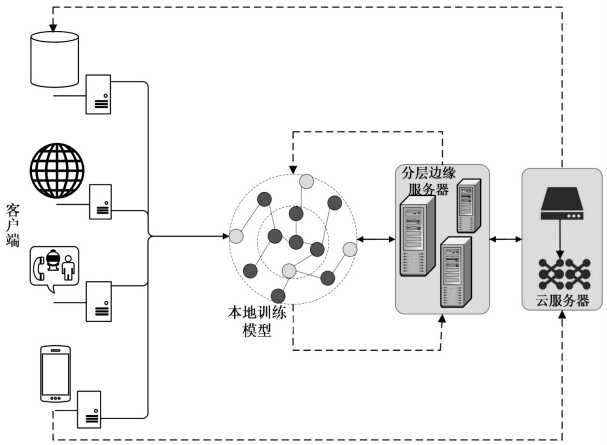

图 1 ECDPFL 模型

2.2 本地差分隐私保护

为了保护 ECDPFL 中客户端数据的隐私,深度神经网络由云、LESs 和客户端协作训练,可以实现 LDP。从数学的角度来看,让 x 表示客户端局部模型中某一层的输出,而 $\sigma(x)$ 表示向 x 添加校准噪声的函数。如果对于所有相邻的输入对 x 和 x',以及所有的输出 M 集,都存在以下不等式,则称输出 $y=\sigma(x)$ 满足 ε -差分隐私。

$$p_r[\sigma(x) \in S] \leqslant e^{\varepsilon} \in p_r[\sigma(x') \in S] \tag{7}$$

为了在 LDP 环境下保护隐私,可以通过引入校准噪声来扰动本地客户端模型的权值。设 w 表示原始权值向量,w' 表示满足 ε -LDP 的摄动权值向量。将拉普拉斯分布通过文献[15]得到噪声的摄动权值向量。

$$w' = w + \delta \tag{8}$$

其中,δ 是一个比例参数 c 的拉普拉斯分布的噪声向量。比例参数 c 是根据权重的灵敏度和期望的隐私参数 ε 来确定的。

$$c = \frac{\theta_\omega}{\varepsilon} \tag{9}$$

其中,θ_ω 表示权重的敏感系数,捕获了当任何单个个体的数据被修改时权重向量的最大可能变化。它可被定义为:

$$\theta_\omega = \max_{w,w'} \| w - w' \|_1 \tag{10}$$

其中,$\| \bullet \|_1$ 表示 L1 范数。将拉普拉斯噪声纳入 L1 规范增加了随机变化,增强了个人贡献的隐私。噪声水平由隐私预算 ε 决定,较低的 ε 值可以提供更强的隐私保护,但可能会降低准确性。

3 实验模拟与评估

3.1 实验设置

在一台配备了 10 核 CPU、8 核 GPU 和 16 GB 内存的服务器上进行了实验。系统是使用 PyTorch 1.10.0 构建的,使用 Python 中的 time.sleep() 函数模拟服务器-客户端传输延迟。为了评估该系统,考虑了客户端集合($k \in 20,40,60,80$)和 LES 集合($m \in 4,8,12,16$)的不同组合。每个 LES 授权相同数量的客户使用相同数量的培训数据。在涉及数据分布的场景中,每个客户端被分配了两个不同的样本标签,每个标签包含 400 个样本,以确保非 IID 数据分布。

3.2 实验数据处理

在对 ECDPFL 框架的评估中,使用常用的 CIFAR-10 和 MNIST 数据集进行图像分类任务,以确保评估的最佳选择。数据集按照 6∶1∶3 比例划分为训练集、测试集和验证集。对于 CIFAR-10 数据集,使用了一个具有 5 852 170 个参数的 CNN 模型。它包括一个输出层和 3×3 卷积层,并使用 32×32 像素图像的 50 000个样本进行训练。类似地,对于 MNIST 数据集,使用一个具有 21 840 个参数的 CNN 模型,用 60 000个图像测量为 28×28 像素的 10 分类手写图像样本进行训练。两个数据集进行本地计算的初始学习率 $\eta=0.01$,衰减指数学习率 $\varphi=0.995$,随机梯度下降动量 $\tau=0.5$,两个数据集不同的隐私预算分别是 $\varepsilon_1=0.5$(MNIST)和 $\varepsilon_2=5$(CIFAR-10),并且每一批次大小设置为 32 个。以此,为提出的 ECDPFL 模型提供一个健壮的评估框架。此外,考虑了现有的三个基线模型,包括传统的 FL[16]、SFL[5]和分层 FL(HFL)[17],并与其进行对比实验。实验采用准确率(Accuracy)和损失值(Loss)作为评价指标,其计算公式如下:

$$ACC = (TP + TN)/(TP + TN + FP + FN) \tag{11}$$

$$\text{Loss} = \text{frac}\{1\}\{m\}/\text{sum\_}\{i=1\}^{\{m\}}(\text{hat}\{y\}^{\{(1)\}} - y^{\{(i)\}})^2 \quad (12)$$

其中, $y^{(i)}$ 表示第 i 个样本的模型预测值, m 表示样本数量。

3.3　实验结果分析

在 CIFAR-10 上使用 ResNet18 将 ECDPFL 与 FL、SFL 和 HFL 进行了比较,LESs 从 4 到 16,客户端从 20 到 80。在固定次数的迭代后评估训练的准确性, p_1 设置为 5, p_2 设置为 2。在大多数情况下,ECDPFL 优于其他方法,在 16 个 LESs 和 80 个客户中达到了 80.12% 的准确率,超过了 FL、SFL 和 HFL,如表 1 所示。然而,随着网络复杂性的增加,对 FL 和 HFL 客户端的计算需求也在增加。虽然 SFL 的服务器客户端方法有所帮助,但它可能会使云服务器使用更多的客户端。因此,ECDPFL 的分层设计优化了网络拓扑结构,并在这种情况下改进了学习能力。然而,在 4 边服务器和 20 个客户端设置中,经过 100 次迭代,HFL 实现了比 SFL 更好的效果,这强调了网络拓扑、数据集属性和算法设计在学习结果中的作用。此外,FL 和 SFL 的性能在不同的网络配置中也有所不同。例如,在 8 个边服务器和 40 个客户端中,SFL 在使用 ResNet18 进行 100 次迭代后达到了 63.10% 的准确率,而 FL 仅达到 62.53%。在 12 个 LESs 和 60 个客户中,FL 在准确率上优于 SFL,为 79.75%,而 SFL 的准确率为 58.99%。

表 1　不同客户端编号对 CIFAR-10 精度的影响

模型	Accuracy/%			
	4LESs-20 clients	8LESs-40 clients	12LESs-60 clients	16LESs-80 clients
FedAvg	59.88	58.23	61.18	60.88
SFL	59.96	63.10	58.99	61.41
HFL	76.68	62.53	79.75	79.80
ECDPFL	**77.28**	**78.66**	**79.85**	**80.12**

使用与 CIFAR-10 评估中相同的客户端和 LES 计数,在四种场景中使用 MNIST 和 ResNet50 测试了 ECDPFL。如表 2 所示,ECDPFL 在所有场景中的精度都优于 FL、SFL 和 HFL,100 次迭代后的最大增益为 1.8%,最小增益为 0.8%。在大多数情况下,它也显示出更快的收敛速度。在准确性方面,ECDPFL 分别达到 90.68%、91.22%、91.80% 和 92.82%,而 HFL 以 91.75% 和 92.78% 位居第二。FL 和 SFL 在所有场景下记录的准确率都低于 90%。这凸显了 ECDPFL 和 HFL 是最有效的快速收敛方法。表 3 比较了损失值的收敛速率。在 4 个边、20 个客户端设置中,ECDPFL 在 100 次迭代后实现了最快的收敛速度,损失了 0.28。HFL 紧随其后,损失为 0.52,而 SFL 和 FL 的损失更高。在 16 个边、80 个客户端配置中,ECDPFL 保持了领先地位,HFL 紧随其后,进一步说明了它们的优越性。

表 2　不同客户端编号对 MNIST 精度的影响

模型	Accuracy/%			
	4LESs-20 clients	8LESs-40 clients	12LESs-60 clients	16LESs-80 clients
FedAvg	85.09	86.66	88.03	89.11
SFL	85.63	86.71	88.90	89.96
HFL	89.91	90.99	91.75	92.78
ECDPFL	**90.68**	**91.22**	**91.80**	**92.82**

表 3　MNIST 数据集上的损失值比较

模型	Loss/%	
	4LESs-20 clients	16LESs-80 clients
FedAvg	1.72	0.96
SFL	0.70	0.77
HFL	0.52	0.49
ECDPFL	**0.28**	**0.41**

在表 4 中,说明了通过隐私预算 ε 量化的隐私和 ECDPFL 方法在 CIFAR-10 和 MNIST 数据集上的训练精度之间的显著权衡。ε 值的选择是有意的,CIFAR-10 选择较大的值(1、2 和 5),而 MNIST 选择较小的值(0.1、0.2 和 0.5)。这种区别是由数据集的特征驱动的;CIFAR-10 具有更复杂和更敏感的图像数据,受益于更大的 ε 值,以提高模型的精度。相比之下,更简单的 MNIST 数据集需要更强的隐私保护,因此 ε 值更小。这些 ε 的选择有助于在该领域内进行有意义的比较,强调了该方法对不同隐私需求的适应性。值得注意的是,在没有 LDP 的情况下,不应用 ε 的训练可以达到最高的准确性,但以牺牲隐私为代价。反转 ε 值可能会以牺牲隐私为代价提高 CIFAR-10 的准确性,而 MNIST 可能会更倾向于隐私而不是准确性。这种理解增强了 ECDPFL 和其他方法在不同隐私场景之间有意义的比较。

表 4　隐私预算在不同数据集上的影响

模型	Accuracy/%	
	CIFAR-10	MNIST
FedAvg	69.73	86.15
SFL	72.99	88.47
HFL	81.22	94.31
ECDPFL	**83.05**	**97.68**

ECDPFL 在 CIFAR-10 和 MNIST 的训练速度上始终优于 HFL 和 SFL。其优势来自于 ECDPFL 有效地平衡客户机工作负载和通过 LESs 最小化聚合计算。随着全局聚集轮数的增加,ECDPFL、HFL 和 SFL 之间训练时间的差异更加明显。例如,在 25 轮比赛中,ECDPFL 完成训练需要 3.50 千秒,而 HFL 和 SFL 分别需要 3.69 千秒和 3.86 千秒(见表 5)。这一趋势也适用于 MNIST,这证实了 ECDPFL 是在这些数据集上进行训练的一个更有效的框架。

表 5 隐私预算在不同数据集上对聚合时间的影响

模型	全局聚合轮数/千秒							
	CIFAR-10				MNIST			
	10	15	20	25	10	15	20	25
FedAvg	*				*			
SFL	1.73	2.63	3.25	3.86	1.63	2.30	3.01	3.62
HFL	1.72	2.55	3.10	3.69	1.62	2.24	2.98	3.50
ECDPFL	**1.68**	**2.42**	**2.92**	**3.50**	**1.51**	**2.14**	**2.74**	**3.33**

4 结束语

该文提出一种新的 ECDPFL 框架,该框架解决了 LES 和 FL 环境中的挑战,帮助拥有有限资源的客户参与模型培训。为了实现使资源有限的客户能够为模型培训做出贡献的目标,ECDPFL 利用多个 LESs 进行部分模型聚合,同时最小化训练时间和能量消耗。经验评估也揭示了该框架在平衡通信和计算方面相对于已建立的两层 FL 架构的优势。值得注意的是,在同步过程中加入 LDP 是为了增强局部模型参数的保密性。在未来的工作中,自适应隐私预算将被集成到 ECDPFL 中,从而在本地客户模型中添加 LDP 噪声时降低计算成本。总之,将差分隐私技术和联邦学习技术结合应用于边缘计算场景对隐私保护[18]是一种有效的保护手段。

参考文献:

[1] 温木奇,温武少. 边缘计算的安全挑战与解决方法综述[J]. 计算机系统应用,2024,33(11):38-47.

[2] NATH S,WU J. Deep reinforcement learning for dynamic computation offloading and resource allocation in cache-assisted mobile edge computing systems[J]. Intelligent and Converged Networks,2020,1(2):181-198.

[3] WANG X,HAN Y,WANG C,et al. In-edge AI:intelligentizing mobile edge computing,caching and communication by federated learning[J]. IEEE Network,2019,33(5):156-165.

[4] DENG Y,LYU F,REN J,et al. Share:shaping data distribution at edge for communication-efficient hierarchical federated learning[C]//2021 IEEE 41st international conference on distributed computing systems (ICDCS). Los Alamitos:IEEE,2021:24-34.

[5] THAPA C,ARACHCHIGE P C M,CAMTEPE S,et al. Splitfed:when federated learning meets split learning[C]//Proceedings of the AAAI conference on artificial intelligence. [s. l.]:AAAI,2022:8485-8493.

[6] TURINA V,ZHANG Z,ESPOSITO F,et al. Combining split and federated architectures for efficiency and privacy in deep learning[C]//Proceedings of the 16th international conference on emerging networking experiments and technologies. Barcelona:SIGCOMM,2020:562-563.

[7] LIU X,DENG Y,MAHMOODI T. Wireless distributed learning:a new hybrid split and federated learning approach[J]. IEEE Transactions on Wireless Communications,2022,2(6):121-129.

[8] 张海超,李嘉坤,刘 东,等. 体联网环境下的边缘智能协同隐私保护方案[J]. 信息技术,2024(8):127-133.

[9] YANG Z,CHEN Y,HUANGFU H,et al. Robust split federated learning for u-shaped medical image networks[J]. arXiv:2212.06378,2022.

[10] 胡海峰,张 熙,赵海涛,等. 移动边缘计算中通信高效的联邦学习模型剪枝算法[J]. 物联网学报,2024,8(20):1-20.

[11] SHI L,SHU J,ZHANG W,et al. Hfl-dp:hierarchical federated learning with differential privacy[C]//Proc IEEE int. glob. commun. conf. (GLOBECOM). Madrid:IEEE,2021:1-7.

[12] ZHOU T. Hierarchical federated learning with Gaussian differential privacy[C]//2023 9th international conference on computer and communications (ICCC). Chengdu:ICCC,2023:22-30.

[13] WAINAKH A,GUINEA A S,GRUBE T,et al. Enhancing privacy via hierarchical federated learning[C]//Workshop IEEE European symp. security privacy (EuroSPW). [s. l.]:IEEE,2020:344-347.

[14] CHANDRASEKARAN V,BANERJEE S,PERINO D,et al. Hierarchical federated learning with privacy[C]//2022 IEEE global communications conference (GLOBECOM). Madrid:IEEE,2022:361-375.

[15] WU N,PENG C,NIU K. A privacy-preserving game model for local differential privacy by using information-theoretic approach[J]. IEEE Access,2020,8:216741-216751.

[16] LI X,HUANG K,YANG W,et al. On the convergence of fedavg on non-iid data[J]. arXiv:1907.02189,2019.

[17] LIU L,ZHANG J,SONG S,et al. Client-edge-cloud hierarchical federated learning[C]//ICC 2020-2020 IEEE international conference on communications (ICC). Dublin:IEEE,2020:1-6.

[18] 孙剑明,赵梦鑫. 边缘计算下差分隐私的应用研究综述[J]. 计算机科学,2024,51(S1):896-904.